International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-12, October 2019

Hybrid Invasive Weed Optimization with Greedy Algorithm for an Energy and Deadline Aware Scheduling in Cloud Computing

PradeepVenuthurumilli, Sridhar Mandapati

Abstract—Cloud computing is being envisioned to be the computing paradigm of the next generation primarily for its advantages of on-demand services, risk transference, resource pooling that is independent of location and ubiquitous network access. Service quality is allocated using various resources in the scheduling process. The deadline refers to the time period from task submission until task completion. An algorithm that has good scheduling attempts at keeping the task executed inside the constraint of the deadline. The Genetic Algorithm (GA) is a common metaheuristic that is used often in literature for procuring solutions that are either optimal or near-optimal. The Invasive Weed Optimization (IWO) is an evolutionary algorithm that is population-based with certain interesting specifications like creations of offspring that are based on the levels of fitness of the parents which increases the size of the population and generates new population by making use of the best among parents and the best among off-springs. The Greedy Algorithms will construct an object that is globally best by means of continuously choosing the option that is locally the best. In this work, a hybrid GA with the Greedy Algorithm and a Hybrid IWO with the Greedy Algorithm that has been proposed for the energy and the deadline-aware scheduling in cloud computing.

Keywords: Invasive Weed Optimization (IWO), Cloud computing, Genetic Algorithm (GA), Greedy Algorithms.

I. INTRODUCTION

Today, distributed system is used in cloud computing by providing dynamically scalable and flexible resources of computing at a low cost. It is now the buzzword in computing services of high performance since it provides access which is on-demand to the shared resource pool over the Internet. It is also in the form of self-service which is metered and is dynamically scalable. However, cloud computing still remains in its infancy and in order to get more benefits, a lot of research is needed in an array of topics. A very important issue of research was for it to be focused in the efficiency of performance especially in scheduling.

For cloud computing, every job will need some resources for completing the task of the user such as the Two-Stage Scheduling model of Deadline Aware.

Revised Manuscript Received on October 30, 2019.
* Correspondence Author
PradeepVenuthurumilli*, Research Scholor, AcharyaNagarjuna University, Guntur, A.P, India. (E-mail: pradeepvenuthuru@gmail.com)
Dr. Sridhar Mandapati, Associate Professor, RVR & JC Engineering College, Chowdavaram, Guntur, A.P, India.

All resources used in the cloud will normally be Virtual machines and a scheduler tends to schedule n number of requests for jobs and will further assign the necessary resource of the cloud (the VMs) for every such request [1].

Two-Stage Scheduling Deadline Aware a scheduler is used by the different users to assign the job based on scheduling. Two types of virtual machines are used in one task to perform specific task. The deadline for the workflow has been distributed over all the tasks and every such task will receive a particular portion of its deadline on the basis of the VM which is one of the most cost-efficient aspects of the task.

The main intent of the scheduling procedure is to detect the resources which are suitable for the system. NP-Hard problems are obtained in the Cloud computing scheduling falls which occur owing to the size of the solution space being large which can result in a long time involved in identifying an optimal solution. There have been hardly any algorithms that are capable of producing optimal solutions inside of polynomial time for solving these issues. In the cloud environment, it may be better to identify a new suboptimal solution within a shorter time frame. The techniques based on metaheuristics are capable of achieving solutions that are near-optimal within a reasonable time frame for these problems.

The techniques based on metaheuristics [2] are concerned with these issues which are done by providing solutions that are near-optimal inside a reasonable time frame. Today, metaheuristics have been gaining popularity owing to an effective and efficient manner in which complex and large problems can be solved. The Genetic Algorithm (GA) [3], is a method of random search and it simulates the process of biological evolution which is used widely in the resource scheduling of cloud computing. Genetic Algorithm has been used owing to its parallel, efficient and global traits of search. But, a traditional GA also has certain disadvantages like slow speed of convergence and local convergence.

Another approach that is extremely well-suited for the cloud resource environments that are heterogeneous that are very dynamic and connected to the schedule of the process is the Greedy Algorithm. This is found well-suited for the dynamic and heterogeneous environment of research which is connected to a scheduler by means of an environment of homogeneous communication. The Greedy Algorithm generally makes a choice which appears to be the best at that time. For improving the task completion in the Greedy Algorithm and the time taken for it, it aims at minimizing turnaround tasks for individual tasks as this can improve the overall time taken for computation. The Invasive Weed Optimization (IWO) algorithm has been inspired by the colonizing of weeds that were used in order to mimic the behaviour of weeds in nature that occupy ideal places for its growth and reproduction.

This has randomness, adaptation, and robustness aside from being simple and effective in obtaining an accurate ability for global search. There are several hybrid algorithms [4] that have been developed by means of combining either two or more such algorithms for improving or enhancing the overall efficiency of search. Researchers normally make an attempt at using all the merits of the individual algorithms towards a common good. Some algorithms may perform well by providing results that are competitive whereas other algorithms may not be able to perform efficiently. The trail and errors are obtained in the system by using hybrid algorithms. Thus, hybridization can prove to be a metaheuristic method that is evolutionary. Developing hybrids that are better will require understanding and insight of all essential parts of the algorithm. This is dependent on the experience and skill of the developer of the algorithm. In case it is possible to analyse the algorithms, it is possible to determine their superiorities. One or more part is added to the algorithm to boost up the system.

For this work, there is a hybrid GA-Greedy along with a hybrid IWO-Greedy Algorithm which was employed for deadline and energy-aware scheduling. The remainder of the investigation was organized in the following manner. The work which was related in the literature was explained and discussed in Section 2. All methods employed were discussed in Section 3. The results of the experiment were reviewed in Section 4 and Section 5 concluded the investigation.

II. LITERATURE SURVEY

Kumaresan and Venkatesan [5] had proposed a hybrid heuristic-based HEFT in order to provide efficient scheduling of tasks in a cloud environment. Additionally, the concept of duplication of the task will be combined with the procedure of Greedy Randomized Search for solving problems of optimization in an environment that is heterogeneous. The results of the experiment have shown that the algorithms proposed could bring down the time and cost to a significant level.

Basically, an pay as you go option is used in the cloud computing process to provide efficient service to the end user. In cloud environment both carbon dioxide and data centres are released by consuming the energy. Hence pollution is reduced by reducing the energy. A new green computing solution is introduced by Kar et al [6]. This system will reduce the pollution. Energy-Aware Task Scheduler with the Genetic Algorithm is also used to minimise the energy consumption.

Different scheduling algorithms [2] is introduced by the Kalra and Singh. This is mainly used in grid and cloud environments. Different techniques have been introduced they are given as Ant Colony Optimization (ACO), the Genetic Algorithm (GA), Particle Swarm Optimization (PSO).

Wang et al [7] presented the Improved Greedy Genetic Algorithm (IGAA)which was based on a base point for generating an initial population that was good and for combining the hybrid algorithms to obtain an optimal solution. It was tested using Traveling Salesman Problem (TSP), with results that demonstrated the algorithm to be effective and feasible in solving complex problems of optimization.

Cloud computing also provided storage and computing services with high scalability. Scheduling was responsible for choosing the resources that were best-suited taking parameters that were both static and dynamic into consideration. The current deadline constrained application only met the deadline need only if critical and there was no special incentive to complete the application faster. Mathiyalagan et al [8] had further introduced another new model in task scheduling for the data centre of cloud computing for energy-efficient and dynamic task scheduling. The Dynamic Power-Aware Greedy Scheduling based algorithm (DPAGS) was a heuristic algorithm that dynamically estimated the task energy by taking into consideration certain other factors which included demands of task resources, the efficiency of VM power, the workload of servers even before scheduling the tasks in a manner which was Greedy. This simulated a VM cluster which was heterogeneous and also conducted an experiment for evaluating its effectiveness. Results of simulation proved the DPAGS was able to reduce effectively the total consumption of energy by 20% and did not result in scheduling overheads that were high. Finally, this simulation was conducted and its efficiency was analysed compared to the other algorithms in existence.

The concept of task scheduling in the environment of cloud computing is a primary concern that has to be addressed for improving performance and for increasing satisfaction of customers. Even though there have been several algorithms of task scheduling, the current approaches focus on the minimization of time taken for completion. Furthermore, managing its Quality of Service (QoS) is very challenging. There was a novel algorithm called the MGGS (Modified Genetic Algorithm (GA) with the Greedy Strategy) that was proposed by Zhou et al [9] for leveraging the modified GA with the Greedy Strategy for optimization of the process of scheduling. For evaluating the final performance of the MGGS, there was a comparison made to the existing algorithms on the parameters of time taken for completion, QoS and average response time. The results proved the MGGS to have performed well in comparison to the other algorithms of task scheduling.

Dong et al [10] had further introduced another new model for task scheduling in the data centre of cloud computing for analysing task scheduling which were energy-efficient. For this, there were several assignments were formulated to the servers which was a problem of integer-programming that had the objective of minimization of the energy consumed by servers in the data centre.

For this, it was proved that the greedy search scheduler will bound the service of constraint at the same time minimizing the active servers. Taken as a practical approach, an efficient-server-first task-scheduling for minimizing consumption of energy of the data centre servers was proposed.

The most-efficient-server-first based schedules tasks to servers of a minimum number and keep the response time of the data centre inside the maximum constraint. The stability of the most-efficient-server-first scheme which was for the tasks that were distributed exponentially, identically distributed and independent arrivals was proved. The results of simulation proved the consumption of server energy for the most-efficient-server-first scheduling scheme that was proposed to be 70 times below the scheme of random-based task-scheduling.

III. METHODOLOGY

The optimization of the cloud system to bring down the cost of operation, increasing efficiency of energy and meeting deadlines of users as defined in their Service Level Agreements (SLAs) has been addressed from the point of view of CSP in this work. Optimized techniques are proposed based on GA, IWO and greedy for minimizing its average makespan and maximizing the utilization of resources under constraints of a deadline. In the models proposed, the tasks are sorted on the basis of the priority of length and the labelling of the state of the VM as successful to attain the deadline. Based on the list, the tasks are assigned toappropriate VMs achieving minimum time of processing. The section further details the Greedy Algorithm, the Proposed IWO with the Greedy Algorithm, the Proposed GA with the Greedy Algorithm and the IWO.

a. Greedy Algorithm

The Greedy algorithm [11], when solving a problem will make a choice which at that time seems to be ideal. This means, it is not generally optimal and the entire one will be considered along with what it has been the best only at the time of a local and optimal solution. The Greedy approach used for an optimized profit has been one of the best approached employed for determining task scheduling issues. The algorithm known as the greedy deadline resource allocation is explained as below:

1. An input virtual machine will be the input.
2. For every resource found in the resource cache, it has to be checked if the current resource which is in a suspended state or in a waking state. If it is so, identify the remaining resource capacity to be checked.
3. In case a resource is in a sleeping state, identify the remainder of resource capacity.
4. Process this function to obtain a resource from a resource cache

```
Greedy (D.n)
// D is a domain from which a solution is to be obtained of size n
// Assume
Solution = 0
For i< 1 to n do
{
S = select (D)
If feasible (Solution, S)) then
        Solution = Union (Solutions, s)
}
Return Solution
```

The Greedy algorithms tend to be advantageous over the other algorithms which are: Simplicity: the Greedy Algorithms are easier to be described and coded than other algorithms. Efficiency: they are implemented in a more efficient manner compared to other algorithms.

b. Proposed Genetic with Greedy Algorithm

The GA had been introduced in the year 1975 by Holland which represented a method of optimization that was based on the evolution process metaphor which was observed in nature [2]. For this, there was a fitness function that was defined for checking the chromosome and its suitability of the environment. Based on the fitness value, there were chromosomes that were chosen and the operations of crossover and mutation were performed for producing the offspring of the new population. The evaluation of the quality of the fitness functions for each offspring was further performed. This process will be repeated until such time all necessary offspring are created. There are advantages for both of the approaches, where the greedy algorithm was incorporated within the framework to the Genetic Algorithm. The technique aims at combining a dependable and effective method of local search for the greedy algorithms used currently that have robust capabilities of global search. The flowchart for the Genetic Algorithm is depicted in figure 1.

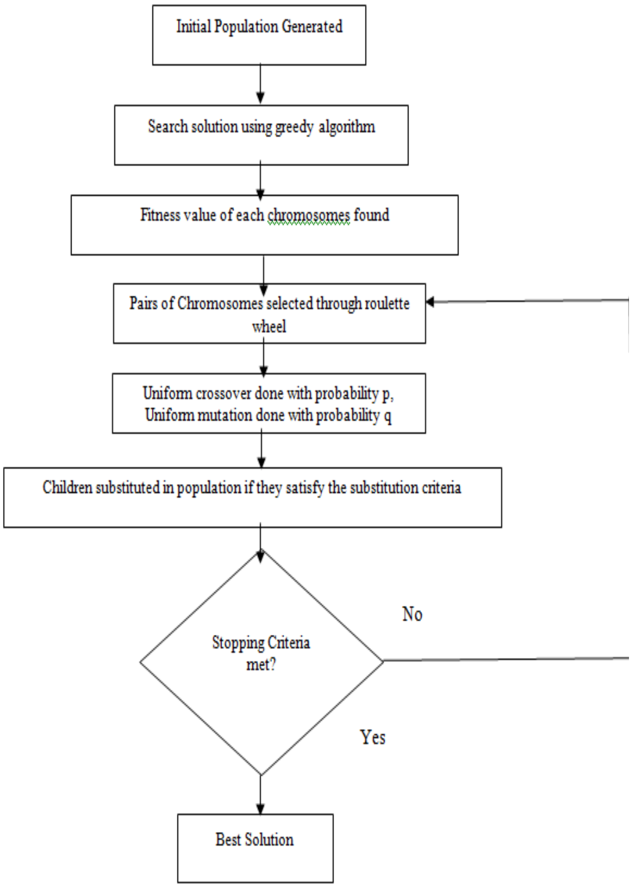

Fig. 1: Flowchart of the Hybrid Genetic Algorithm (GA) with Greedy Algorithm

c. Invasive Weed Optimization (IWO)

The Invasive weed Optimization had been developed in the year 2006 by Mehrabian and Lucas which was an optimization algorithm that was numerical and bio-inspired for simulating the natural weed behaviour when finding an ideal place for reproduction and growth. Evolutionary algorithms are introduced to compare the properties of various distinctive [12]. The IWO begins by initializing the population which means it is generated randomly in the problem space. Depend on the fitness of population seeds are produced.

A by from equation (2), the relation of non linear simulation and alteration is observed:

$$\sigma_{cur} = \frac{(iter_{max} - iter)^n}{(iter_{max})^n}(\sigma_{init} - \sigma_{final}) + \sigma_{final} \qquad (2)$$

In which, an $iter_{max}$ is with maximum iterations, denotes all maximum iterations, σ_{cur} the standard deviation which was in the current time step and n denotes a nonlinear modulation index. All produced seeds, when accompanied by parents were taken as potential solutions for the subsequent generations. Lastly, there was a competitive exclusion that has been conducted and once a certain number of iterations are complete, the population will reach its maximum for which an elimination has to be employed. For this, both seeds and parents will be ranked together and the ones that have a better level of fitness will become reproductive.

d. Proposed Invasive Weed Optimization with Greedy Algorithm

The basic Greedy Algorithm principle is: it begins with an initial solution to a problem for approaching a set goal for obtaining better solutions in minimal time. If it is not able to proceed further, it stops. The Greedy Algorithm was found to be effective in solving problems of optimal substructures which means local optimum that determines its global optimum. The problem is divided into sub-problems which is recurred to its optimum for a final problem.

A new hybrid IWO greedy algorithm is introduced by combining all algorithms. The execution procedure of this algorithm is given below:

- Firstly, candidate solutions (C) will be initialized after which an empty set is initialized.
- Next, the Greedy Algorithm is executed and if the candidate solution does not equal an empty set S, it is not a solution. For this, x is chosen from set C with a greedy strategy. In case {x} union S proves to be a feasible solution S returns as the solution.
- There may be a tendency of jumping to the next step which performs the IWO algorithm. As soon as the seeds can find their position, they get ranked with parents (as a weed colony).
- After this, the weeds that have lower fitness levels get eliminated for reaching the maximum allowable population for a colony. This way, both seeds, and weeds were ranked together with the ones that have better fitness surviving and replicating.
- By compared with previous one, a new memeplex is obtained. Compared to previous one, this will give effective result.
- Now, the global best gets swapped with the greedysolution.
- Lastly, in case the criterion of termination is met, its ultimate best candidate solution is taken to be the output. If not, all subsequent iterations will begin from the memeplex algorithm again.

IV. RESULTS AND DISCUSSION

In this section, GA-Greedy and IWO-Greedy methods are used. Experiments are carried out using 5000 to 20000 number of tasks. Table 1 and 2 shows the parameters of GA and IWO respectively. The guarantee ratio, energy savings and resource utilization as shown in tables 3 to 5 and figures 1 to 3.

Table 1 Parameters of GA

Number of individuals	100
Number of Generations	200
Crossover type	Multipoint crossover
Crossover rate	0.72
Mutation Rate	0.004
Selection Strategy	Roulette wheel selection
Length of individual chromosome	8
Number of variables	3

Table 2 Parameters of IWO

Problem dimension	9
Maximum number of Iterations	180
Minimum number of seeds	1
Maximum number of seeds	5
Initial value of standard deviation	60
Final value of standard deviation	10

Table 3 Guarantee Ratio % for proposed IWO-Greedy
IWO-Greedy

Number of Tasks	GA-Greedy	IWO-Greedy
5000	88	91
7500	95	95
10000	94	95
12500	93	94
15000	95	97
17500	96	99
20000	96	99

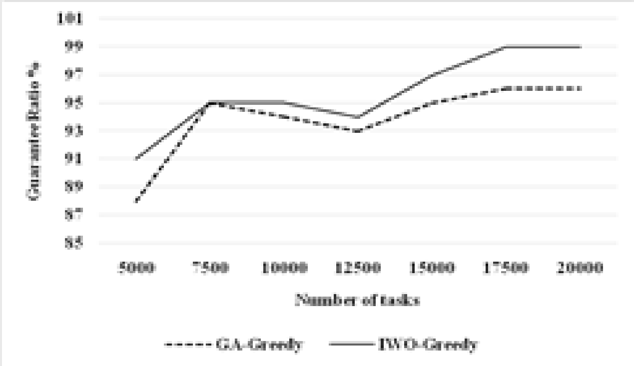

Fig. 2: Guarantee Ratio % for proposed IWO-Greedy

Higher guarantee ratio % is obtained in the proposed IWO greedy system that is 3.35%. this percentage is obtained for 5000 number of tasks. In the same way 1.06% is obtained for the 75000 number of tasks and 2.1%is obtained for the 15000 number of tasks. At last 3.08% is obtained for GA greedy environment by using 2000 number of tasks.

Table 4 Energy Savings % for proposed IWO-Greedy

Number of Tasks	GA-Greedy	IWO-Greedy
5000	2.9	3.2
7500	3.5	4
10000	3.2	4.3
12500	3.1	4
15000	3.5	3.9
17500	3	3.7
20000	2.7	3.2

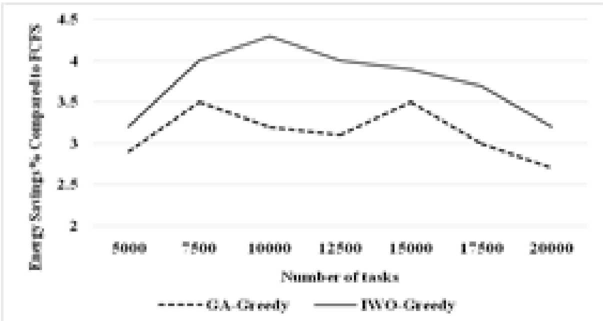

Fig. 3: Energy Savings % for proposed IWO-Greedy

Higher energy % is obtained in the proposed IWO greedy system that is 9.84%. This percentage is obtained for 5000 number of tasks. In the same way 13.33% is obtained for the 75000 number of tasks , 25.35% is obtained for 12500 number of tasks, 29.33% is obtained for 10000 number of tasks, 10.8% is obtained for the 15000 number of tasks and20.89% is obtained for 2000 number of tasks . At last 16.95% % is obtained for GA greedy environment by using 2000 number of tasks.

Table 5 Resource Utilization % for proposed IWO-Greedy

Number of Tasks	GA-Greedy	IWO-Greedy
5000	62	65
7500	69	73
10000	79	82
12500	78	82
15000	81	84
17500	81	85
20000	80	86

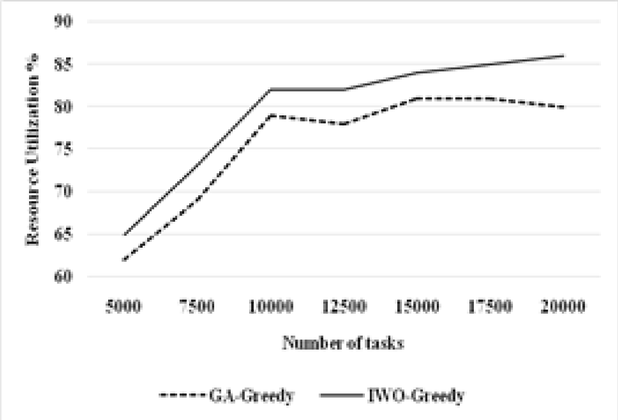

Fig. 4: Resource Utilization % for proposed IWO-Greedy

Higher Resource Utilization % is obtained in the proposed IWO greedy system that is 4.72%. This percentage is obtained for 5000 number of tasks. In the same way 5.63% is obtained for the 75000 number of tasks , 5%is obtained for 12500 number of tasks, 3.73% is obtained for 10000 number of tasks, 3.64% is obtained for the 15000 number of tasks and 4.82% is obtained for 2000 number of tasks . At last 7.23% is obtained for GA greedy environment by using 2000 number of tasks.

V. CONCLUSION

Cloud computing today has become a new paradigm of computing bringing an unparalleled level of flexibility. The metaheuristic algorithms do have a strong place in the field of research owing to their high efficiency in solving several problems in cloud computing. The IWO is an optimization technique with derivative-free real parameters mimicking the colonizing weeds and their ecological behaviour. The Greedy algorithm is a fast and simple algorithm since it chooses the solution that has been described in a greedy criterion. The results have proved that the IWO-Greedy which was proposed has a higher ratio of guarantee percentage by about 3.35% for the 5000 number of tasks, the same value for the 7500 number of tasks, by about1.06% for the 10000 number of tasks, by about1.07% for the 12500 number of tasks, by about2.1% for the 15000 number of tasks, by about3.08% for the 20000 number of tasks, and finally by about 3.08% for the 20000 number of tasks on being compared to the GA-Greedy.

REFERENCES

1. Raju, I. R. K., Varma, P. S., Sundari, M. R., & Moses, G. J. (2016). Deadline aware two stage scheduling algorithm in cloud computing. *Indian Journal of Science and Technology*, *9*(4).
2. Kalra, M., & Singh, S. (2015). A review of metaheuristic scheduling techniques in cloud computing. *Egyptian informatics journal*, *16*(3), 275-295.
3. Huang SC, Jiau MK, Lin CH (2015) A genetic-algorithm-based approach to solve carpool service problems in cloud computing. IEEE Trans IntellTranspSyst 16(1):352–364.
4. Kaveh, A., Talatahari, S., &Khodadadi, N. (2019). Hybrid Invasive Weed Optimization-Shuffled Frog-Leaping Algorithm for Optimal Design of Truss Structures. *Iranian Journal of Science and Technology, Transactions of Civil Engineering*, 1-16.
5. Kumaresan, M., &Venkatesan, G. P. (2016, December). Cloud Scheduling Using Hybrid Heuristic Based HEFT and Enhanced GRASP Approach: A Study and Analysis. In *2016 8th International Conference on Computational Intelligence and Communication Networks (CICN)* (pp. 48-51). IEEE.
6. Kar, I., Parida, R. R., & Das, H. (2016, March). Energy aware scheduling using genetic algorithm in cloud data centers. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)* (pp. 3545-3550). IEEE.
7. Wang, Z., Duan, H., & Zhang, X. (2009, August). An improved greedy genetic algorithm for solving travelling salesman problem. In *2009 Fifth International Conference on Natural Computation* (Vol. 5, pp. 374-378). IEEE.
8. Mathiyalagan., P, Rajshree., M, Shobika., S and Sowmiya., M (2017) “An Efficient Dynamic Scheduling based on Greedy Algorithm for Heterogeneity Clouds”, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 6, Issue 3, March 2017.
9. Zhou, Z., Li, F., Zhu, H., Xie, H., Abawajy, J. H., &Chowdhury, M. U. (2019). An improved genetic algorithm using greedy strategy toward task scheduling optimization in cloud environments. *Neural Computing and Applications*, 1-11.
10. Dong, Z., Liu, N., & Rojas-Cessa, R. (2015). Greedy scheduling of tasks with time constraints for energy-efficient cloud-computing data centers. *Journal of Cloud Computing*, *4*(1), 5.
11. Kumar, D., &Mandal, T. (2016, April). Greedy approaches for deadline based task consolidation in cloud computing. In *Computing, Communication and Automation (ICCCA), 2016 International Conference on* (pp. 1271-1276). IEEE.
12. Lenin, I., Reddy, B. R., &Kalavathi, M. S. (2014). Hybrid-invasive weed optimization particle swarm optimization algorithm for solving optimal reactive power dispatch problem. *Int. J. Res. Electron. Commun. Technol. (IJRECT 2014)*, *1*(1), 41-45.